

Pembuatan Game Arcade Dengan Kendali Tongkat dan Deteksi Suara

Daniel Stevanus, Nana Ramadijanti, S.Kom, M.Kom, Rizky Yuniar H.

Mahasiswa D4LJ Jurusan Teknik Informatika, Dosen Politeknik Elektronika Negeri Surabaya

Politeknik Elektronika Negeri Surabaya Institut Teknologi Sepuluh Nopember

Kampus ITS, Keputih Sukolilo, Surabaya 60111

Telp. (+62)-31-5947280 Fax. (+62)-31-5946114

ABSTRAK

Game *arcade* dengan kendali tongkat dan deteksi suara ini merupakan suatu aplikasi game yang dikembangkan dari berbagai aspek teknologi yang dikemas menjadi sebuah program. Diantaranya adalah aspek *Human Computer Interaction* (HCI) , *Image Processing* dan *Speech Recognition*. Tujuan utama aplikasi ini adalah untuk menciptakan game yang mampu menimbulkan terjadinya interaksi antara pemain dengan game itu sendiri.

Proses utama pada game ini dibagi menjadi tiga tahap, yakni pendeteksian tongkat, pendeteksian suara dan sistem game. Metode *Haar Training* digunakan untuk pendeteksian tongkat, untuk mendeteksi suara, kita memanfaatkan *Speech API* pada *library windows*. Sedangkan pada proses pembuatan game itu sendiri menggunakan *Visual C++* pada *dotnet*.

Kata Kunci : *Speech API, OpenCV*

ABSTRACT

Arcade games with the control stick and the speech recognition is a game which developed by various aspects of technology packed into single program. There are include *Human Computer Interaction* (HCI), *image processing* and *speech recognition*. The main purpose of this application is to create a game that could make a interaction between the players with the game itself.

The main processes on this game is divided into three stages, there are detection of the stick, the speech recognition and gaming systems. *Haar training* methods used for detection of the stick, to speech recognition, we take advantage of *API's Speech* at the *library windows*. While in the game development process itself using *Visual C++* on *dotnet*.

Keyword : *Speech API, OpenCV*

1. PENDAHULUAN

Semakin maraknya game interaktif yang memungkinkan pengguna dapat berinteraksi langsung di dalam game, seperti pada konsol game terkemuka

Nintendo-Wii. Memunculkan ide dari para pembuat game untuk menciptakan game baru yang sesuai dengan para imajinasi gamers (pengguna game) itu sendiri.

Seperti kita ketahui nama Harry potter identik dengan penyihir yang mampu mengeluarkan jurus – jurusnya dengan melambatkan tongkat nya, hal ini memunculkan ide untuk membuat sebuah game dengan menggunakan kendali tongkat, sebagai contoh apabila kita mengayunkan tongkat dan mengucapkan perintah suara maka object pada game akan bereaksi, Sensor dari game ini berasal dari Web Cam sebagai input yang akan mendeteksi ada atau tidaknya gerakan dari tongkat dan microphone sebagai input dari beberapa perintah yang diberikan dengan kata yang telah ditentukan dengan memanfaatkan api *library* pada *windows*.

1.1 Tujuan

Tujuan umum dari pembuatan proyek akhir ini adalah membuat game yang interaktif dan atraktif sehingga mengajak user untuk lebih kreatif dan berpikir dalam bermain game. Sedangkan tujuan khusus dari pembuatan proyek akhir ini adalah untuk membuat aplikasi yang dapat mendeteksi arah gerakan tongkat dan deteksi suara dengan cepat dan tepat. Sehingga diperoleh kondisi yang *real time* seperti dengan menggunakan keyboard atau mouse.

1.2 Batasan Masalah :

Adapun beberapa hal yang membatasi pemakaiannya dalam proyek akhir ini, antara lain :

- Satu pemain.
- Kata yang diucapkan hanya terbatas dan sudah tersedia di game.
- Bentuk tongkat harus seperti yang telah ditentukan.
- Intensitas cahaya harus cukup
- Jarak kamera dengan objek terbatas.
- Tidak ada dominan suara yang lain.

2. TEORI PENUNJANG

Pada bab ini akan membahas mengenai teori-teori yang akan dijadikan materi penunjang dalam pembuatan proyek akhir ini. Adapun teori-teori yang akan dibahas di dalam bab ini adalah :

2.1 Pengolahan Citra

Pengolahan citra adalah suatu metode yang digunakan untuk mengolah gambar sehingga menghasilkan gambar yang sesuai dengan keinginan kita. Pengambilan gambar bisa dilakukan dengan menggunakan kamera atau alat lain yang bisa digunakan untuk mentransfer gambar misalnya scanner atau kamera digital. Bahasan kali ini berfokus pada pengambilan gambar menggunakan web kamera. Sehingga citra yang dihasilkan sudah berbentuk sinyal digital dan mudah dikenali atau dibaca oleh komputer. Citra digital adalah citra kontinyu yang sudah didiskritkan baik koordinat spasial maupun kecerahannya. Citra digital dianggap matrik dengan ukuran $M \times N$ dimana baris dan kolom menunjukkan titik-titiknya. Citra berwarna menggunakan metode RGB. Adapun masing-masing warna dalam tabel memiliki 3 buah kombinasi angka yaitu R, G, dan B yang menentukan proporsi warna merah, warna hijau dan warna biru dari warna tersebut. RGB masing-masing memiliki range antara 0 hingga 63 sehingga jumlah warna yang dapat kita pilih untuk mengisi warna pada sebuah cell ditabel ialah $63 \times 63 \times 63 = 16$ juta warna. Tetapi seluruh tabel hanya dapat diisi dengan 256 pilihan warna.

2.2 Pengertian Gesture

Dalam studi pustaka, Andy Wilson dan Aaron Bobick dalam makalahnya menjelaskan tentang cara kerja gerak tangan manusia dimana gerak tangan manusia setiap gerakannya mengalami moment-moment tertentu misalnya pada saat seseorang tersebut istirahat maupun menggerakkan tangannya. Isyarat dari setiap interaksi nonverbal, menggunakan bahasa gerak baik wajah, tangan, kaki maupun otot-otot yang lainnya sebagai alat untuk berinteraksi dengan komputer.

2.3 Teknik Pengambilan Citra

Permasalahan awal dari pengolahan citra ialah bagaimana cara kita menangkap gambar dari keadaan real untuk kemudian di sampling sehingga terbentuk suatu gambar atau citra digital. Permasalahan ini biasa di sebut dengan *capture*. Proses ini memerlukan alat-alat *capture* yang baik seperti kamera, scanner, light-pen dan lain sebagainya, sehingga kita bisa mendapatkan hasil yang baik pula.

2.4 Contour Finding

Contour sebenarnya adalah sebuah nilai sequences, maka segala sesuatunya harus berada pada koor sequences. Contour ditemukan jika terdapat perbedaan titik-titik yang tinggi dengan tetangganya. Hal ini terjadi karena didasarkan pada hasil deteksi tepi. Maka metode ini kemungkinan dapat mempunyai error yang lumayan tinggi, apabila dihadapkan dengan berbagai macam obyek yang ada di dalam suatu image/gambar. Oleh karena itu, harus diberi batasan bahwa dalam suatu image/gambar hanya boleh terdapat satu obyek, terutama dalam gambar/image yang akan menjadi templatnya.

2.5 Object Detection

Metode yang digunakan untuk melakukan *object detection* pada proyek akhir ini adalah *Haar Classifier*, yaitu metode object detection yang membangun sebuah boosted rejection cascade, yang akan membuang data training negative sehingga didapat suatu keputusan untuk menentukan data positif. *Haar Classifier* merupakan metode supervised learning, yaitu membutuhkan data training untuk dapat mendeteksi obyek-obyek tertentu. Untuk itu, *Haar Classifier* membutuhkan data positif (obyek yang akan dideteksi) dan data negatif (bukan obyek yang akan dideteksi).

2.6 OpenCV Library

OpenVC adalah suatu library gratis yang dikembangkan oleh developer-developer Intel Corporation. Library ini terdiri dari fungsi-fungsi computer vision dan image processing tingkat tinggi. OpenCV sangat disarankan untuk programmer yang akan berkuat pada bidang computer vision, karena library ini mampu menciptakan aplikasi yang handal, kuat dibidang digital vision. Karena library ini bersifat cuma-cuma dan sifatnya yang open source, maka dari itu OpenCV tidak dipesan khusus untuk pengguna arsitektur Intel, tetapi dapat dibangun pada hampir semua arsitektur.

Saat ini para developer dari Intel Corporation telah membuat berbagai macam versi, yaitu:

- openCV untuk bahasa pemrograman C/C++,
- openCV untuk bahasa pemrograman C# (masih dalam tahap pengembangan), dan
- openCV untuk bahasa pemrograman Java.

untuk bahasa pemrograman C# dan Java, karena masih dalam tahap pengembangan, maka kita membutuhkan library lain sebagai pelengkap kekurangan yang ada. Namun untuk bahasa pemrograman C/C++ tidak memerlukan library lainnya untuk pemrosesan pada computer vision.

2.7 Sistem Pengenalan Pembicaraan

1. **Mode diktasi.** Pada mode ini pengguna komputer dapat mengucapkan kata/kalimat yang selanjutnya akan dikenali oleh komputer dan diubah menjadi data teks. Kemungkinan jumlah kata yang dapat dikenali dibatasi oleh jumlah kata yang telah terdapat pada database. Pengenalan mode diktasi merupakan speaker dependent. Keakuratan pengenalan mode ini bergantung pada pola suara dan aksan pembicara serta pelatihan yang telah dilakukan.

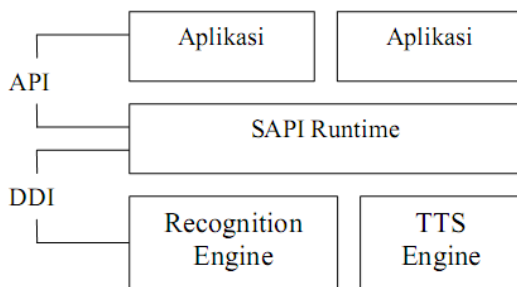
2. **Mode command and control.** Pada mode ini pengguna komputer mengucapkan kata/kalimat yang sudah terdefinisi terlebih dahulu pada database dan selanjutnya akan digunakan untuk menjalankan perintah tertentu pada aplikasi komputer. Jumlah

perintah yang dapat dikenali tergantung dari aplikasi yang telah mendefinisikan terlebih dahulu pada database jenis-jenis perintah yang dapat dieksekusikan. Mode ini merupakan speaker independent karena jumlah kata yang dikenali biasanya terbatas sekali dan ada kemungkinan pembicara tidak perlu melakukan pelatihan.

SAPI 5.1 terdiri dari 2 antar muka yaitu application programming interface (API) dan device driver interface (DDI).

Application Programming Interface(API).

Pada sistem pengenalan pembicaraan, aplikasi akan menerima even pada saat suara yang diterima telah dikenali oleh engine.



Gambar 2.1 Blok Diagram Arsitektur SAPI

3. PERANCANGAN SISTEM

3.1 Instalasi dan Konfigurasi OpenCV

OpenCV adalah suatu library computer vision yang dibuat oleh para developer Intel Corporation yang bersifat Open Source, library tersebut merupakan kumpulan fungsi-fungsi C dan beberapa kelas C++ yang mengimplementasikan banyak algoritma-algoritma Image Processing dan Computer Vision.

Modul-modul OpenCV, antara lain:

- cv* – fungsi-fungsi utama OpenCV,
- cvaux* – fungsi tambahan (eksperimental) OpenCV,
- cxcore* – dukungan struktur data dan aljabar linear,
- highgui* – fungsi-fungsi yang menangani masalah GUI.

3.2 Instalasi Speech API SDK

Speech API adalah suatu library computer vision yang dibuat oleh para developer Windows, library tersebut merupakan kumpulan fungsi-fungsi C dan beberapa kelas C++ yang mengimplementasikan banyak algoritma-algoritma speech processing .

Proyek akhir ini memakai Speech SDK versi 5.1 .Untuk instalasi hanya seperti install program seperti biasa.Kemudian Speech digunakan di aplikasi speech yang menggunakan VB.NET kemudian

digabungkan ke aplikasi game yang menggunakan VC++.NET .

3.3 Perancangan Sistem Game

3.3.1 Perancangan Sistem image Processing

Perancangan sistem ini dimulai dari training objek tongkat menggunakan algoritma Haar cascade yang di lakukan dengan menggunakan OpenCV, objek tongkat yang digunakan sudah didefinisikan terlebih dahulu dengan menggunakan tongkat yang spesifik warna dan bentuknya, perancangan deteksi ini secara murni menggunakan tool yang telah disediakan oleh Haar Cascade.

Pengolahan citra dilakukan dengan training image pada algoritma haar training. Dengan keterangan sebagai berikut:

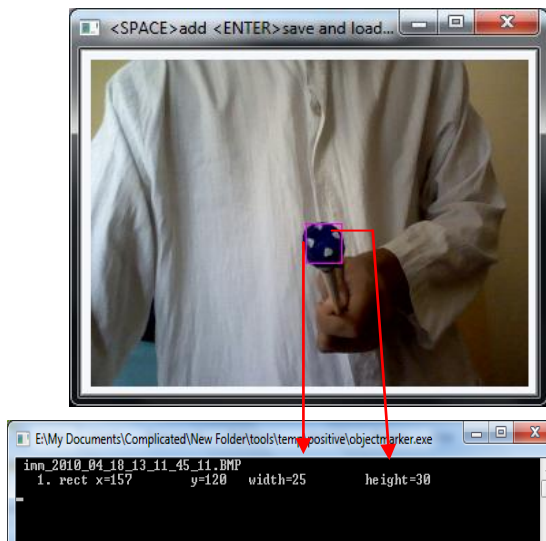
Mempunyai 1800 data gambar positif masing-masing objek, yang dimaksud gambar positive adalah gambar yang didalamnya terdapat gambar suatu objek yang akan dideteksi. Seperti, gambar tongkat. Kemudian mempunyai 2000 gambar negatif ,yang dimaksud dengan gambar negative adalah suatu gambar yang didalamnya tidak terdapat suatu objek yang akan dideteksi dan gambar tersebut bertipe grayscale. Kemudian dilakukan crop gambar objek pada gambar positif sesuai dengan objek yang akan dideteksi.

Setelah keseluruhan proses cropping selesai maka akan didapatkan nilai vector gambar tersebut, kemudian dilakukan proses training haar. Taining haar ini memerlukan waktu yang cukup lama, berdasarkan jumlah gambar dan jumlah stage yang terdapat pada training. Semakin banyak jumlah gambar dan semakin besar nilai stage, maka hasil deteksi objek yang di dapatkan juga semakin baik. Setelah training ini selesai, maka akan didapatkan nilai variable cascade. Nilai cascade tersebut kemudian dikonvert menjadi xml database. dengan menggunakan nilai Xml tersebut pada program. Maka program akan dengan mudah mendeteksi objek yang terdapat pada gambar.



Gambar 3.1 Gambar Positif

Selanjutnya dari sekumpulan gambar positif tersebut dilakukan data bounding atau cropping terhadap objek spesifik yang akan dideteksi, sebagai contoh gambar positif yang telah di bounding sebagai berikut.



Gambar 3.2 Proses crop dan bounding pada data positif.



Gambar 3.3 Gambar Negatif

3.3.2 Deteksi Objek

Setelah training selesai maka xml akan menjadi suatu database objek deteksi sudah. Kamera akan menangkap gambar kemudian aplikasi melakukan load database pada xml yang sudah terbuat maka aplikasi akan mencari objek yang akan dideteksi.

Aplikasi akan mengcapture image dari webcam hasil capture tersebut akan melakukan konversi warna RGB ke Gray. selanjutnya gambar/image grayscale tersebut dirubah skalanya.

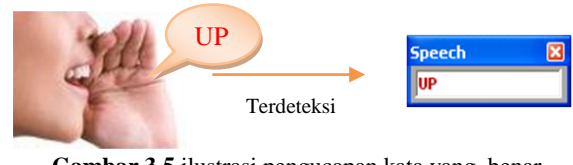


Gambar 3.4 Hasil deteksi tongkat

3.4 Speech Recognition

Speech Recognition dilakukan oleh library SpeechAPI (SAPI) SDK 5.1 dan dibuat di aplikasi yang berbeda menggunakan VB.NET. Dalam pendeteksian suara proyek akhir ini akan memanfaatkan SAPI menjadi dalam bentuk perintah suara yang telah ditentukan.

Suara akan diucapkan melalui microphone dan akan diproses untuk mencari mana kata yang paling sesuai diantara kata-kata yang ditentukan.



Gambar 3.5 ilustrasi pengucapan kata yang benar



Gambar 3.6 ilustrasi pengucapan kata yang salah

3.5 Penggabungan Speech ke Aplikasi Game

Pada proyek akhir ini, pembuatan game dilakukan menggunakan bahasa pemrograman VC++ sedangkan proses speech recognition dibuat menggunakan bahasa pemrograman VB.net. Maka untuk menyatukan kedua program ini agar bisa bekerja bersinambungan diperlukan suatu jembatan. Disini kita akan memanfaatkan suatu file teks yang berfungsi untuk menampung kata yang telah diucapkan oleh user. Dari kata yang ada pada file teks tersebut bisa dimanfaatkan untuk proses implementasi pada Game. Apabila kata pada teks berhasil di baca, maka selanjutnya kata – kata tersebut akan di gunakan untuk pengkondisian suatu perintah pada game.

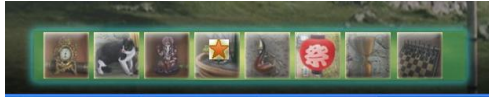
3.6 Implementasi System Pada game

Setelah deteksi objek tongkat dan speech recognition berhasil selanjutnya akan digabungkan dengan sistem pada game. Deteksi objek tongkat berfungsi sebagai pengganti mouse dalam menggerakkan pointer pada game, pointer ini digunakan untuk mengambil dan meletakkan objek gambar yang di inginkan, sedangkan speech recognition berfungsi sebagai perintah suara untuk mengkondisikan apakah objek akan di ambil atau di taruh.



Gambar 3.7 Pointer mengikuti gerakan tongkat

Untuk mendeteksi apakah pointer sudah berada di atas objek yang diinginkan atau tidak. Maka di buat segmentasi tiap objek, sehingga tiap satu objek memiliki definisi sendiri. Hal ini akan mempermudah dalam mendeteksi apakah pointer berada di atas objek atau tidak



Gambar 3.8 Pointer berada di atas sebuah objek gambar

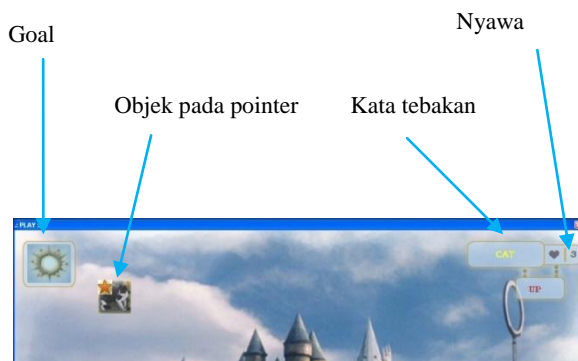


Gambar 3.9 Pointer berada di atas sebuah objek gambar

Apabila pointer telah berada di atas objek gambar, selanjutnya game akan mengkondisikan apakah gambar tersebut akan dibawa oleh pointer atau tidak sesuai dengan perintah suara “UP”.

Apabila kondisi objek gambar telah terbawa oleh pointer dan pemain telah membawa pointer menuju ke sasaran, maka objek gambar tidak akan dieksekusi selama tidak ada perintah suara “DOWN”. Perintah ini hanya berlaku pada area tujuan atau *goal* dan berfungsi untuk proses check apakah objek yang dibawa sesuai dengan kata yang muncul pada game.

Untuk membatalkan objek yang dibawa, dapat menggunakan perintah “NO”. Jika objek yang dibawa pointer sesuai dengan kata yang muncul. Maka objek tersebut akan menghilang dan kata random yang lain akan muncul yang mengindikasikan bahwa pemain benar dalam menebak gambar. Sedangkan bila objek gambar tersebut tidak sesuai dengan kata yang muncul. Maka pemain telah salah dalam menebak selanjutnya objek gambar tersebut akan dikembalikan ke posisi semula dan nyawa atau *life* dari pemain akan berkurang satu.



Gambar 3.9 Objek harus sesuai dengan kata tebakan

4. ANALISA DAN PENGUJIAN

Pengujian pada proyek akhir ini meliputi pengujian speech recognition, deteksi tongkat hingga

keseluruhan jalannya game. Hal ini dimaksudkan untuk menguji tingkat keberhasilan sistem dalam kondisi tertentu, misalkan: kondisi suara bising atau kondisi ruangan yang kurang akan pencahayaan. Sehingga dapat di peroleh analisa pada tiap masalah yang muncul.

4.1 Pengujian Image processing

Pengujian ini terdiri dari :

- Pengujian terhadap cahaya
- Jarak tongkat dengan kamera
- Posisi tongkat
- Kecepatan gerakan tongkat

Kesimpulan pengujian yaitu semakin banyak cahaya di sekitar ruangan. Maka menghasilkan deteksi tongkat yang sangat baik, jarak terbaik antara tongkat dengan kamera adalah berjarak 10 – 25 cm. posisi tongkat paling mudah terdeteksi pada posisi 0° dan 45°. Sedangkan bila posisi tongkat di tegakkan secara 90°, tongkat tidak akan terdeteksi.

4.2 Pengujian Speech

Terdiri dari :

- Keras Lemahnya Suara
- Kebisingan Area Pemain
- Kualitas Microphone Pemain

Kesimpulan pengujian yaitu diharapkan suara diucapkan dengan keras, area tidak bising, dan kualitas microphone bagus karena dapat mengurangi noise.

5. KESIMPULAN

Setelah melaksanakan pengujian dan analisa pada system “Pembuatan Game arcade Dengan Gerakan Tangan dan Perintah Suara”, maka dapat diambil beberapa kesimpulan sebagai berikut:

- Proses deteksi tongkat sangat baik pada area dengan intensitas cahaya yang cukup
- Jarak maksimal yang masih mampu dideteksi adalah 40 cm, selebihnya system tidak dapat mendeteksi adanya tangan. Dan jarak terbaik bagi proses pendeteksian adalah 10 – 25 cm.
- Untuk mendapatkan hasil yang optimal dalam pendeteksian tangan, maka posisi tongkat harus dalam keadaan sudut 0°.
- System mampu mengenali obyek tongkat dengan baik jika tongkat berada dalam keadaan diam atau bergerak dalam keadaan lambat.
- Suara pemain lebih baik diucapkan dengan keras.
- Area pemain diusahakan sepi tidak ada noise disekitarnya.
- Microphone berpengaruh melantunkan kata dengan menggunakan microphone, maka suara pemain akan lebih mudah dideteksi dan terhindar dari noise suara yang lain.

DAFTAR PUSTAKA

1. Open computer vision library.
<http://sourceforge.net/project/opencvlibrary/>.
2. Tom Cassey, Tommy Chheng, Jefery Lien, "Camera Based Hand and Body Driven Interaction with Very Large Display", ucsd lab, June 15, 2007.
3. Neo, Naotoshi. "OpenCV Haar Training". 2007.
4. Sun, Min. "OpenCV and face detection". 2006.
5. Gary Bradski, Adrian Kaehler. "Learning OpenCV". Oreilly team's Intel. 2005.
6. Paul Viola and Michaels J. Jones. "Rapid Object Detection using boosted Cascade of Simple Features". IEEE CVPR, 2001.
7. Robin Hewitt. "seeing with openCV part 2 – Finding Face in image". Jan, 2007.
8. Gary Agam. "introduction to programming with OpenCV". 2006.
9. Ardiansyah Rizkcy. "Kendali Game Tetris Menggunakan Gerakan Tangan", EEPIS-ITS. 2009.
10. Yusuf. "Pembuatan Game Puzzle Gambar Dengan Gerakan Tangan dan Perintah Suara", EEPIS – ITS. 2010.
11. Microsoft Speech API 5.1.
<http://www.microsoft.com/speech/developers.aspx>.
12. Harrington, M, (2006) 'Giving Computers a Voice',
<http://msdn.microsoft.com/coding4fun/inthebox/tts-hw/default.aspx>.